FIG. 1

FIG. 2

Project - Form1 (Code) ～200

(General)	▽	ChangeFont

```
Private Sub ChangeFont()
Dim mytext As TextBox
```
⎱201

210 → ～211
～202
mytext ⎵ ～212 ～231 ～223 240

230 →

221～	🗏 ⌐Alignment ↙ ⌐	△
222～	🗏 Appearance	▭ ～242
223～	🗏 BackColor	～220
224～	🗏 BorderStyle	
225～	🗏 Container	
226～	🗏 DataChanged	
227～	🗏 Datafield	▽

～241

FIG. 3

Project - Form1 (Code) ～200

(General)	▽	ChangeFont

```
Private Sub ChangeFont()
Dim mytext As TextBox
```

210 →
212～ ～420
211 → mytext ⎸f ⎸ ～202 ～430

410～	🗏 Font	△
411～	🗏 FontBold	▭
412～	🗏 FontIItalic	
413～	🗏 FontName	
414～	🗏 FontSize	～220
415～	🗏 FontStrikethru	
416～	🗏 FontUnderline	▽

FIG. 4

Project - Form1 (Code) — 200

(General) ▽ ChangeFont

```
Private Sub ChangeFont()
Dim mytext As TextBox
```

210 →

510 511
mytext.font\ | — 202

531 — 🌊	┊Bold┊
532 — 🌊	Charset
533 — 🌊	Italic
534 — 🌊	Name
535 — 🌊	Size
536 — 🌊	Strikethrough
537 — 🌊	Underline

530

FIG. 5

Project - Form1 (Code)

(General) ▽ ChangeFont

```
Private Sub ChangeFont()
Dim mytext As TextBox
```

210 →

511 610 611
mytext.font⌐Underline=

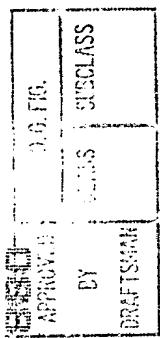|	
641 — 🌊	┊False┊
642 — 🌊	True

640

FIG. 6

Project1 - Module1 (Code)

(General) | ▽ | MyProc

```
Enum MyColor
    blue
    green
    red
End Enum
```
710

```
Sub MyProc(Name As String, c As MyColor, Optional i As Integer = 2)
```
721 722 723 720

```
    . . .
End Sub
```

```
Sub MainProc()
    MyProc|
```
731 733

732

730

MyProc(*Name As String*, c As MyColor, [*i As Integer = 2*])

741 742 743 740

FIG. 7

700

700

☐ Project1 - Module1 (Code)

(General)   ▷ │ MyProc

```
Enum MyColor
    blue
    green
    red
End Enum

Sub MyProc(Name As String, c As MyColor, Optional i As Integer = 2)
    . . .
End Sub

Sub MainProc()
    MyProc "Smith"
```

810

511

733

732

MyProc(*Name As S*|Color, [*i As Integer = 2*])

741   742   743

740

850

851 blue
852 green
853 red

FIG. 8

Project1 - Module1 (Code)

(General) ▽ | MyProc

```
Enum MyColor
    blue
    green
    red
End Enum


Sub MyProc(Name As String, c As MyColor, Optional i As Integer = 2)

· · · ·

End Sub


Sub MainProc()
    MyProc "Smith", blue, |5
```

910 911 912

MyProc(Name As String, c As MyColor, [i As Integer = 2])

741 742 743

732

740

700

FIG. 9

FIG. 10

START — 1008

BEGIN PROGRAMMING
LANGUAGE EDITOR — 1010

1020   1030

⊗ 1015

1021
ARE
AUTOMATIC
FEATURES
ENABLED
?

NO

YES

1031
MANUAL
REQUEST
?

NO

YES

TO
FIG. 11 — 1023

TO
FIG. 13 — 1033

1026
CONTINUE
?

YES

NO

1036
ANOTHER
REQUEST
?

YES

NO

QUIT — 1028

QUIT — 1038

FIG. 11

FROM FIG. 10 STEP 1023 — 1108

BEGIN COMPILE ENGINE — 1110

READ NEXT CHAR/COMMAND — 1118

HOT-KEY OR MENU COMMAND ? — 1125 — YES → TO FIG. 13 — 1132 →

NO ↓

ADMIN COMMAND ? — 1141 — YES → EXECUTE COMMAND — 1145 → UPDATE ASSIST WINDOW — 1149 →

NO ↓

COMMIT KEY ? — 1155 — YES → TO FIG. 12 — 1158 →

NO ↓

NON-CHAR EDIT COMMAND ? — 1160 — YES →

NO ↓

ADD CHARACTER TO PROGRAM STATEMENT — 1172

ANY INFORMATION AVAILABLE ? — 1175 — NO / YES

YES ↓ UPDATE\GENERATE ASSIST WINDOW — 1185

CONTINUE ? — 1188 — YES →

NO ↓

RETURN TO FIG. 10 — 1190

FIG. 12

1200

FROM
FIG. 11
STEP 1158 — 1205

REPLACE WORD IN
PROGRAM STATEMENT
WITH SELECTED
MENU ITEM — 1208

1210

NON-DELIMITER
COMMIT KEY
?

YES → THROW AWAY
COMMIT KEY
CHARACTER — 1220

NO

ADD COMMIT
CHARACTER TO
PROGRAM STATEMENT — 1240

1250

ANY NEW
INFORMATION
TO DISPLAY IN
AN ASSIST
WINDOW
?

NO → REMOVE ANY
EXISTING WINDOW — 1254

YES

UPDATE/GENERATE
ASSIST WINDOW — 1258

RETURN
TO FIG. 11 — 1260

FIG. 13A

1300

1308 FROM FIG. 10 OR FIG. 11

1311 IDENTIFY PROCESSING REQUIRED

1320 ⊗

1330

1331 PARSE PROGRAM STATEMENT INTO TOKENS

1334 LOCATE PROCEDURE ID TOKEN

1335 IS ID A SYMBOL ? → NO / YES

1340

1341 PARSE PROGRAM STATEMENT INTO TOKENS

1344 LOCATE PROCEDURE ID TOKEN

1345 IS ID A SYMBOL ? → NO / YES

1350

1351 LOCATE OPERATOR OR PRECEEDING CURSOR

1353 IDENTIFY OBJECT & MEMBER PRECEEDING OPERATOR

1354 IS TOKEN A SYMBOL ? → NO / YES

1360

1361 PARSE PROGRAM STATEMENT INTO TOKENS

1362 BIND EXPRESSION EXCLUDING LAST IDENTIFIER

1365 GENERATE MEMBER LIST FROM RETURNED SYMBOL

IDENTIFY PRESENT ARGUMENT TOKEN — 1336

IS PRESENT ARGUMENT A SYMBOL ? — 1337

NO

YES

GENERATE ASSIST WINDOW FOR LIST OR CONSTANT VALUE — 1338

IDENTIFY PRESENT ARGUMENT TOKEN — 1346

GENERATE ASSIST WINDOW MAP FOR PROCEDURE CALL — 1347

DETERMINE OBJECT TYPE AND MEMBER RETURN TYPE — 1355

IS LAST METHOD AN ENUM ? — 1356

NO

YES

GENERATE ASSIST WINDOW WITH MEMBER LIST — 1357

GENERATE ASSIST WINDOW OF POSSIBLE CHOICES — 1358

RETURN TO FIG. 10 OR FIG. 11 — 1370

FIG. 13B